Centre de recherche sur les transports (C.R.T.)
Centre for Research on Transportation

CRT-2003-06

A HEURISTIC ALGORITHM FOR THE POINT OF PRESENCE DESIGN PROBLEM IN IP NETWORKS

by

Steven Chamberland
Marc St-Hilaire
Samuel Pierre

February 2003

Université de Montréal
C.P. 6128, succursale Centre-ville, Montréal QC H3C 3J7 Canada
Téléphone: (514) 343-7575 / Télécopieur: (514) 343-7121

crt@crt.umontreal.ca
www.crt.umontreal.ca

HEC ÉCOLE POLYTECHNIQUE MONTRÉAL Université de Montréal

103-07098/

PUBLICATION
CRT-2003-06

A HEURISTIC ALGORITHM FOR THE POINT OF PRESENCE DESIGN PROBLEM IN IP NETWORKS

by

Steven Chamberland[1,2]
Marc St-Hilaire[2]
Samuel Pierre[2]

This research was supported in part by the Natural Sciences and Engineering Research Council of Canada (NSERC).

1. Centre de recherche sur les transports, Université de Montréal, C.P. 6128, succursale Centre-ville, Montréal, Canada H3C 3J7 (steven.chamberland @polymtl.ca)

2. Computer Engineering Department, École Polytechnique de Montréal, C.P. 6079, succursale Centre-ville, Montréal, Canada H3C 3A7 (samuel.pierre@polymtl.ca)

Abstract

In this paper, we propose a heuristic algorithm for the point of presence (POP) design problem in Internet protocol (IP) networks, where a POP is a node composed of several interconnected co-located routers. This problem consists of selecting the number of routers and their types, selecting the interface card types, connecting the access and the backbone links to the ports and selecting the link types between the co-located routers. A systematic set of experiments is designed to assess the performance of the proposed heuristic algorithm. The results show that quasi-optimal solutions can be found with the proposed algorithm.

Keywords: network planning, point of presence, Internet protocol (IP), mathematical model, heuristic algorithm.

Résumé

Dans cet article, nous proposons une heuristique pour le problème de l'optimisation des points de présence dans les réseaux IP (Internet Protocol). Ce problème consiste à sélectionner le nombre de routeurs et le type de chacun, sélectionner les types de cartes d'interface à insérer dans les routeurs, brancher les liens d'accès et de transmission aux ports et sélectionner les types de liens entre les routeurs co-localisés dans le point de présence. Les résultats numériques montrent que l'heuristique proposée trouve des solutions quasi-optimales.

Mots-clés: planification des réseaux, point de présence, protocole Internet (IP), modèle mathématique, heuristique.

1 Introduction

Internet protocol (IP) networks have grown at an explosive rate during the last decade. As a result, several routers are co-located in the same central office forming large IP point of presence (POP). Considering that IP routers and interface cards are still expensive, POP optimization is an important issue for these providers to remain competitive.

In this paper, we propose a heuristic algorithm for the POP design problem in IP networks. This problem consists of selecting the number of routers and their types (where a type of router is characterized by its number of slots and its switch fabric capacity) to install in the POP, selecting the interface card types (where an interface card type is characterized by its technology, the port rate and the number of ports per card), connecting the access and backbone links to the ports and connecting the network of the co-located routers. The objective is to minimize the cost of the POP. The model also consider the routing of the IP traffic within the POP using a standardized routing protocol (e.g., the open shortest path first (OSPF) routing protocol).

The literature contains a huge amount of articles relating to network planning and optimization, but the planning problem defined here was considered only by Chamberland *et al.* [1]. The authors have proposed a model for the POP design problem and a branch-and-bound algorithm to solve it. However, the results demonstrate that only small-size instances can be solved exactly.

This paper is organized as follows. Section 2 presents the mathematical model for the POP design problem in IP networks, and in Section 3 a heuristic algorithm is proposed. Section 4 presents results for a set of randomly generated problems. Conclusions and further works are presented in Section 5.

2 The Mathematical Model

The following notation is used throughout this paper. The notation is composed of sets, decision variables, cost parameters and constants.

Sets: Let I be the set of clients connected to the POP, J, the set of other POPs connected to the POP and K, the set of potential routers' locations in the POP. Let R be the set of link and port types (where α_r is the bit rate (in bps) of the link/port of type $r \in R$) and T, the set of router types (where m^t is the number of slots in a router of type $t \in T$ and β^t its switch fabric capacity). Finally let L be the set of interface card

types (where n^l is the number of ports on the card of type $l \in L$) such that $L = \cup_{r \in R} L_r$ where L_r is the set of interface cards with ports of type $r \in R$.

Decision Variables: Let u_k^t be a 0-1 variable such that $u_k^t = 1$ if and only if the router installed at location $k \in K$ is of type $t \in T$ and v_k^l, the number of cards of type $l \in L$ installed at location $k \in K$. Let x_{ik}^r be the number of links of type $r \in R^A$ from client $i \in I$ to location $k \in K$, y_{jk}^r, the number of links of type $r \in R^B$ from POP $j \in J$ to location $k \in K$ and $z_{kk'}^r$, the number of links of type $r \in R^P$ installed between location $k \in K$ and location $k' \in K$ (for $k < k'$). Finally, let f_{ik}^o (f_{ki}^o) be the traffic (in bps) from client $i \in I$ to location $k \in K$ (from location $k \in K$ to client $i \in I$) originating from $o \in I \cup J$, f_{jk}^o (f_{kj}^o), the traffic flow (in bps) from POP $j \in J$ to location $k \in K$ (from location $k \in K$ to POP $j \in J$) originating from $o \in I \cup J$ and $f_{kk'}^o$, the traffic flow (in bps) from location $k \in K$ to location $k' \in K$ originating from $o \in I \cup J$.

Cost Parameters: Let a_k^l be the cost of a card of type $l \in L$ and installing it at location $k \in K$, b_k^t, the cost of purchasing a router of type $t \in T$ and installing it at location $k \in K$ and $c_{kk'}^r$, the cost (including the installation cost) of connecting location $k \in K$ to location $k' \in K$ using a link of type $r \in R^P$.

Constants: Let p^{od} be the traffic demand from $o \in (I \cup J)$ to $d \in (I \cup J) \setminus \{o\}$ passing through the POP (in bps) and q_i^r (q_j^r), the number of links of type $r \in R$ from $i \in I$ (from $j \in J$) to the POP.

The model for the POP design problem (PDP) in IP networks is presented below.

PDP:

$$\min \sum_{k \in K} \left(\sum_{l \in L} a_k^l v_k^l + \sum_{t \in T} b_k^t u_k^t + \sum_{r \in R} \sum_{\substack{k' \in K \\ k < k'}} c_{kk'}^r z_{kk'}^r \right) \tag{1}$$

subject to

Client assignment constraints

$$\sum_{k \in K} x_{ik}^r = q_i^r \quad (r \in R, i \in I) \tag{2}$$

POP assignment constraints

$$\sum_{k \in K} y_{jk}^r = q_j^r \quad (r \in R, j \in J) \tag{3}$$

Router-type uniqueness constraints

$$\sum_{t \in T} u_k^t \leq 1 \quad (k \in K) \tag{4}$$

Router-capacity constraints (slot level)

$$\sum_{l \in L} v_k^l \leq \sum_{t \in T} m^t u_k^t \quad (k \in K) \tag{5}$$

Router-capacity constraints (switch fabric level)

$$\sum_{r \in R} \alpha_r \sum_{l \in L_r} n^l v_k^l \leq \sum_{t \in T} \beta^t u_k^t \quad (k \in K) \tag{6}$$

Router-capacity constraints (port level)

$$\sum_{i \in I} x_{ik}^r + \sum_{j \in J} y_{jk}^r + \sum_{\substack{k' \in K \\ k < k'}} z_{kk'}^r + \sum_{\substack{k' \in K \\ k > k'}} z_{k'k}^r \leq \sum_{l \in L_r} n^l v_k^l \quad (r \in R, k \in K) \tag{7}$$

Inter-router network topology constraints

$$\sum_{t \in T} u_k^t + \sum_{t \in T} u_{k'}^t \leq \sum_{r \in R^P} z_{kk'}^r + 1 \quad (k < k', k, k' \in K) \tag{8}$$

Link capacity and flow conservation constraints

Link capacity and flow conservation constraints	(9)

Integrality and nonnegativity constraints

$$\mathbf{f} \in \mathbf{R}_+^{|I||J||K|(|I|+|J|+|K|-1)}, \mathbf{u} \in \{0,1\}^{|K||T|}, \mathbf{v} \in \mathbf{N}^{|K||L|},$$
$$\mathbf{x} \in \mathbf{N}^{|I||K||R|}, \mathbf{y} \in \mathbf{N}^{|J||K||R|}, \mathbf{z} \in \mathbf{N}^{\frac{|K||R|}{2}(|K|-1)}. \tag{10}$$

The objective function (1) is composed of three terms: the cost of the cards, the cost of the routers and the cost of the links. Constraints (2) are client assignment constraints that require each client be connected to the POP with the required access link types and constraints (3) are POP assignment constraints that require each other POPs be connected to the POP with the required backbone link types. The router-type uniqueness constraints (4) impose that at most one router type be installed at location $k \in K$ and constraints (5) necessitate the total number of slots used by the cards at location $k \in K$ be less than or equal to the number of slots available for the router type installed at this location. Constraints (6) impose the sum of the rates of the ports installed to the router at location $k \in K$ be less than or equal to its switch fabric capacity and constraints (7) dictate the number of links of type $r \in R$ connected to a router be less than or equal to the

number of ports of this type. Constraints (8) necessitate that the topology of the inter-router network be the one specified by the network planner. The full-mesh topology is considered in this paper. Constraints (9) are the link capacity and traffic flow conservation constraints (see Chamberland *et al.* [1] for more information on these constraints). Constraints (10) are integrality and nonnegativity constraints.

Since PDP is $\mathcal{NP}$-hard (transformation from the integer knapsack problem [2]), it is unlikely that large-size instances of PDP can be solved to optimality within a reasonable amount of time.

3 The Heuristic Algorithm

In the next section, we propose a heuristic algorithm for PDP, called H-PDP, to find good solutions rapidly.

To describe H-PDP, let an *access router* be a router in the POP used exclusively to connect the access links and a *backbone router*, a router in the POP used to connect the backbone links first and access links only if there are some capacity left in the router. The *size* of the POP is the number of routers in the POP.

H-PDP generates at most $|K|$ solutions corresponding to POP of size n for $1 \leq n \leq |K|$ where $|K|$ is the number of locations to install routers in the POP. For each value of n, H-PDP varies the number of backbone routers b for 1 to n to find the best solution with n routers in the POP. The best solution found is retained by the heuristic.

Algorithm H-PDP

Step 1: (*Initialisation*)

Set $n := 1$ and go to Step 2.

Step 2: (*Find a solution with n routers*)

2.1. Install n routers of maximum capacity in the POP.

2.2. Calculate the minimum number of cards of each type necessary to connect all access and backbone links.

2.3. Set $b := 1$ and go to 2.4.

2.4. (*Find a solution with $n - b$ access and b backbone routers*)

2.4.1. For each $r \in R$, install the minimum number of cards with ports of type r (uniformly among the b backbone routers) to connect the backbone links of type r. If there is not

enough slots and $b < n$ ($b = n$), set $b := b + 1$ and go to Step 2.4 (go to Step 4).

2.4.2. For each $r \in R$, assign the backbone links of type r to the ports of type r.

2.4.3. Compute the traffic between each pair of backbone routers respecting the traffic routing policy.

2.4.4. Install the minimum number of links between each pair of backbone routers to respect the link capacity constraints.

2.4.5. If the capacity constraints of one or more backbone routers are not respected and $b < n$ ($b = n$), set $b := b + 1$ and go to Step 2.4 (go to Step 4).

2.4.6. For each $r \in R$, install the minimum number of cards having ports of type r (uniformly in the $n - b$ access routers) to connect the access links of type r. The backbone routers can be used if there is not enough slots in the access routers. If there is not enough slots (in the access and backbone routers) and $b < n$ ($b = n$), set $b := b + 1$ and go to Step 2.4 (go to Step 4).

2.4.7. For each $r \in R$, assign the access links of type r to the ports of type r in the access routers first.

2.4.8. Compute the traffic between each pair of routers respecting the traffic routing policy.

2.4.9. Install the minimum number of links between each pair of backbone routers to respect the link capacity constraints.

2.4.10. If the capacity constraints of one or more routers are not respected and $b < n$ ($b = n$), set $b := b + 1$ and go to Step 2.4 (go to Step 4).

Step 3: (*Best solution update*)

If the cost of the current solution is less than the cost of the best solution found so far, update this best solution.

Step 4: (*Termination test*)

If $n < |K|$ go to Step 2 with $n := n + 1$. Otherwise, return the best solution found.

Table 1: Characteristics of the router types

Type	A	B
Number of slots	7	14
Switch fabric capacity [Gbps]	20	80
Cost [$]	10 000	20 000

Table 2: Characteristics of the types of interface cards

Type	Technology	Number of ports	Cost [$]
A	OC-3	4	3 000
B	OC-3	16	16 000
C	OC-12	4	10 000
E	OC-48	4	30 000
F	OC-192	1	50 000

4 Numerical Results

In this section, we present a study of the performance of the proposed algorithm H-PDP. The algorithm H-PDP was implemented in the C language on a Pentium 4 PC (1.6 GHz and 512 MB of RAM).

Two link types (i.e., OC-3, OC-12) are used in the access network, two in the backbone network (i.e., OC-12, OC-48) and one in the inter-router network (i.e., OC-192). The costs of the router types are given in Tab. 1 and the costs of the interface card types in Tab. 2. These costs included the purchase and the installation costs (including the floor space, the patch panels, the cables and connectors, the racks, the electric installation and labor). The cost of interconnecting two co-located routers is $500 (including the cables and labor).

Twenty problems were randomly generated as follows. For each client, the number of access links is randomly selected in the interval from 1 to 4 with a maximum of two OC-3 links and two OC-12 links. For each other POP, the number backbone links is randomly also selected in the interval from 1 to 4 with a maximum of two OC-12 links and two OC-48 links. The traffic demand is generated randomly using a uniform distribution law in the interval from 0 to 1 Mbps between each pair of clients, from 0 to 40 Mbps between each pair of other POPs and from 0 to 1 Mbps between each client and other POP.

All heuristic results were compared with a lower bound obtained as follows. The bound for PDP, noted

Table 3: Computational results

$\|I\|$	$\|J\|$	LB	H-PDP		
			Value	CPU	GAP
		[k$]	[k$]	[sec]	[%]
10	10	232.0	236.0	1	1.72
20	10	212.0	222.0	1	4.72
30	10	372.5	382.5	1	2.68
40	10	417.5	428.5	1	2.63
50	10	457.5	464.5	2	1.53
10	15	266.0	266.0	1	0.00
20	15	405.5	422.5	1	4.19
30	15	444.5	458.5	1	3.15
40	15	730.5	755.5	2	3.42
50	15	757.5	765.5	2	1.06
10	20	412.5	416.5	1	0.97
20	20	448.5	462.5	1	3.12
30	20	498.5	502.5	1	0.80
40	20	522.5	522.5	2	0.00
50	20	764.5	775.5	2	1.44
10	25	542.5	546.5	1	0.74
20	25	522.5	526.5	1	0.77
30	25	821.5	839.5	1	2.19
40	25	841.5	855.5	2	1.66
50	25	791.5	805.5	3	1.77

LB (Lower Bound), was obtained by using the CPLEX Mixed Integer Optimizer [3] to solve PDP without the flow variables and constraints (9).

The results are presented in Tab. 3. In this table, the first column presents the number of clients $|I|$ and column 2 the number of other POPs $|J|$. Column 3 presents the lower bound value and columns 4 to 6 present respectively the solution found by algorithm H-PDP, the CPU execution time to find it and the GAP that indicates the percentage gap between the heuristic solution and the lower bound LB.

The following observations can be made from Tab. 3. First, H-PDP finds solutions close to optimality for the problems considered (within 1.93%, on average, from the lower bound) and the maximum CPU time is tree seconds. Moreover, the reader should point out that the heuristic finds the optimal solution two times.

5 Conclusions and Further Works

In this paper, we have proposed a heuristic algorithm for the POP design problem in IP networks.

The heuristic results were compared to a proposed lower bound obtained by relaxing a subset of the original constraints in the model. These results have shown that the proposed heuristic produces quasi-optimal solutions for the 20 test problems considered.

The following directions for further research are open. First, it would be interesting to consider the POP design problem jointly with the access and backbone network design problem. Second, it would be interesting to explore an exact method in order to solve real-size problems to optimality.

References

[1] S. Chamberland, M. St-Hilaire and S. Pierre, "On The POP Optimization Problem in IP Networks," *IASTED Computer Communication Networks*, 2002, pp. 378-384.

[2] M.R. Gary and D.S. Johnson, "Computers and Intractability: A Guide to the Theory of NP-Completeness," New-York, NY: Freeman, 1979.

[3] ILOG Inc., "Using the CPLEX Callable Library and CPLEX Mixed Integer Library," Incline Village: NV, 2000.